

PROJECT REPORT ON ONLINE FEE PAYMENT APPLICATION

SUBMITTED BY:

JOHN MALSAWMTLUANGA HMAR ROLL NO. 2223BCA005

LALMUANZUALA SAILO ROLL NO. 2223BCA011

VANLALAWMPUII..... ROLL NO. 2223BCA018

Under the guidance of:

Mr. H. Lalruatkima

Assistant Professor

**Bachelor of Computer Application
(HATIM)**



MIZORAM UNIVERSITY, TANHRIL: AIZAWL

2024

HIGHER AND TECHNICAL INSTITUTE, MIZORAM

KAWMZAWL, LUNGLEI – 796701



CERTIFICATE

*This is to certify that **John Malsawmtluanga Hmar, Lalmuanzuala Sailo and Vanlalawmpuii** has fully completed the project entitled, “**Online Fee Payment App**” in order to meet the requirement of the Mizoram University for the V Semester Bachelor of Computer Application in the year 2024. It is to certify that all the corrections/suggestions indicated for internal assessment has been incorporated in the project. The project record has been approved as it satisfies the academic requirements in respect of the project work prescribed for the BCA Course.*

(H. LALRUATKIMA)
Project Guide
Dept. of Computer Science

(VUANSANGA VANCHHAWNG)
Principal
HATIM

(K. LALMUANPUIA)
Head of Department
Dept. of Computer Science

Name of Examiner

Signature with date

ACKNOWLEDGEMENT

We begin with our deepest gratitude to God, whose guidance and blessings have been a constant source of strength throughout our project. We are truly thankful for the good health and resilience He has granted us, allowing us to reach this important milestone.

*Our heartfelt appreciation goes to **Mr. H. Lalruatkima**, our dedicated Project Guide, and **Mr. K. Lalmuanpuia**, the esteemed Head of the Department of Computer Science at the Higher and Technical Institute, Mizoram. Your unwavering support, valuable advice, and words of encouragement have been the pillars on which we leaned, giving us the courage to persevere. We are truly grateful for your patience and belief in our abilities; your guidance has been a light that has helped us navigate this journey.*

We also extend our warm thanks to all the wonderful faculty members in the Department of Computer Science. Your kindness, availability, and willingness to assist us whenever we needed it have made this semester a rewarding and unforgettable experience.

Finally, our deepest thanks go to our beloved parents. Your unconditional love, moral support, and endless encouragement have been the foundation upon which we built our determination. We are forever grateful for your constant presence and the confidence you instilled in us every step of the way.

CONTENTS

Fig. No.	Figure Name	Page
1	INTRODUCTION	
	Overview of the Project	
	Objective of the Project	
2	SYSTEM ANALYSIS	
	Hardware Requirement	
	Software Requirement	
	Data Flow Diagram	
3	SYSTEM DESIGN – USER	
	Welcome & Login Page	
	User Registration Form	
	Homepage	
	History Transaction	
	Profile	
4	SYSTEM DESIGN – ADMIN PANEL	
	Admin Home Page	
	Students Table Page	
	Notification Page	
5	CODING – USER	
	Welcome & Login Page	
	User Registration Form	
	Homepage	
	History Transaction	
	Profile	
6	CODING – ADMIN PANEL	
	Admin Home Page	
	Students Table Page	
	Notification Page	

1. INTRODUCTION

CONTENT: Overview of the Project
Objective of the Project

INTRODUCTION

The project name is **FeeBuddy**, and its purpose is to develop an online fee payment application that is planned to run on both Android and iOS platforms and be accessible across all institutions. Another goal of this project is to simplify money transactions; instead of paying fees in cash, they can be paid online, making the process faster and more convenient.

Overview of the Project

FeeBuddy is an innovative online fee payment application designed to simplify and enhance the fee payment process for educational institutions. The project aims to develop a versatile platform that operates seamlessly across Android and iOS devices, catering to the needs of students, parents, and administrators. By shifting fee payment from traditional cash-based methods to a digital format, **FeeBuddy** seeks to provide a more efficient, secure, and accessible solution for managing transactions.

The application is intended to be user-friendly, allowing students and parents to complete payments with minimal hassle. It will also reduce the need for physical visits to institutions and paperwork, enabling a more streamlined approach. **FeeBuddy's** design ensures that users can access the platform at any time and from any location, making it compatible with a variety of devices such as smartphones, tablets, and desktops.

Objective of the Project

1. Simplify Fee Payment Process:

- Develop an intuitive and user-friendly interface for students and parents to pay fees online easily.
- Streamline the entire fee payment process, reducing the need for physical visits and paperwork.

2. Enhance Accessibility:

- Make fee payment accessible from anywhere, at any time, through a web or mobile application.
- Ensure the app is responsive and works well on various devices, including smartphones, tablets, and desktops.

3. Integrate Secure Payment Processing:

- Implement secure payment gateway integration to handle transactions efficiently and safely.
- Ensure compliance with security standards and protect user data during transactions.

4. Automate Invoice Generation and Management:

- Automatically generate invoices for each payment transaction.
- Provide users with easy access to their payment history and downloadable invoices.

5. Provide Real-Time Notifications:

- Implement real-time notifications to inform users about due dates, successful payments, and other important updates.
- Use email or in-app notifications to keep users informed.

2. SYSTEM ANALYSIS

CONTENT: **Hardware Requirement**
Software Requirement
Data Flow Diagram

SYSTEM ANALYSIS

Hardware Requirement

Since FeeBuddy is designed as a versatile online fee payment application accessible to all users, it doesn't have specific or high hardware requirements. The app aims to be functional across a range of devices with basic configurations. The recommended hardware specifications are:

1. Mobile Devices (Android and iOS):

- 🚩 **Operating System:** Android 8.0 (Oreo) or later, iOS 12.0 or later
- 🚩 **RAM:** Minimum 2 GB
- 🚩 **Storage:** 50 MB of free space for installation and updates
- 🚩 **Network Connectivity:** Stable internet connection (3G, 4G, or Wi-Fi)

2. Desktop and Laptops:

- 🚩 **Operating System:** Windows 7 or later, macOS 10.12 (Sierra) or later
- 🚩 **RAM:** Minimum 4 GB
- 🚩 **Storage:** 100 MB of free space for web app caching
- 🚩 **Browser:** Updated versions of Chrome, Firefox, Safari, or Edge
- 🚩 **Network Connectivity:** Stable internet connection (broadband or higher)

3. General Requirements:

- 🚩 **Internet Access:** Consistent and reliable internet connection for smooth payment transactions
- 🚩 **Security Features:** Devices should have basic security measures like antivirus software and firewall protection

Software Requirement

Software requirements for the development of this project are as under: -

1. Integrated Development Environment (IDE): Visual Studio Code is chosen for IDE because it is lightweight and highly customizable. Large selection of extensions for various needs (e.g., linters, debuggers, etc.). It can be used for live share extension for real-time collaborative coding. It has a rich plugin ecosystem.

2. Language: We have chosen Flutter for development because:

- **Dart Language:** Dart is easy to learn and offers good performance, compiling directly to native ARM code.
- **Single Codebase:** Flutter allows us to build apps for iOS, Android, web, and desktop from a single codebase, reducing development time and costs.
- **Performance:** Flutter apps are known for high performance with smooth animations and 60 FPS.
- **Rich UI Components:** Flutter provides a rich set of customizable widgets for Material Design (Android) and Cupertino (iOS) styles.

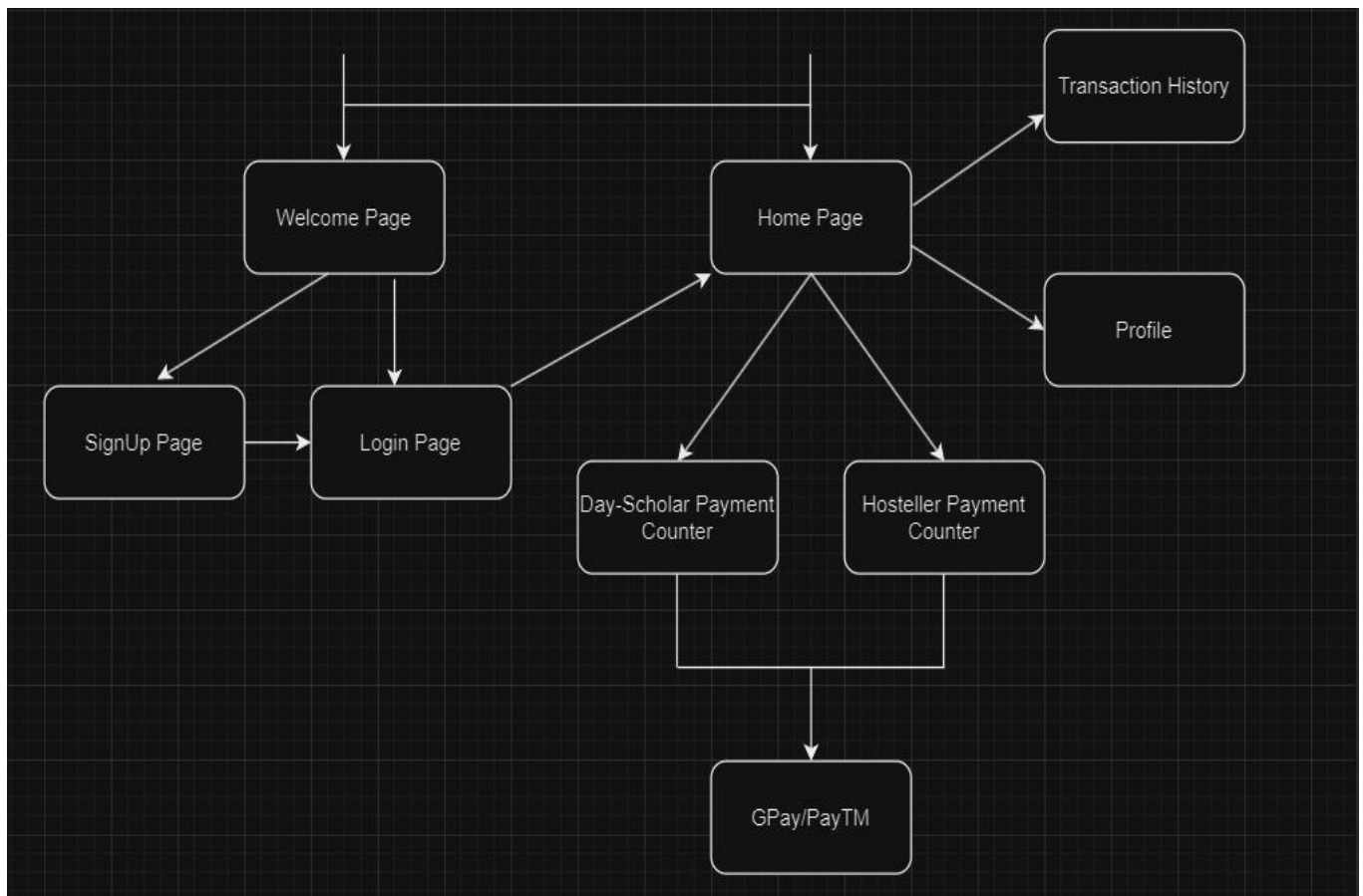
- **Hot Reload:** Instantly view code changes without restarting the app, speeding up development.

3. Database: Firebase offering more powerful querying capabilities. It supports offline data persistence and real-time updates.

4. Payment Gateway Integration: Google Pay(GPay), Pay Through Mobile(PayTM).

5. Notification Service: Firebase Cloud Messaging (FCM) serves as a versatile tool for sending messages and notifications to users across different platforms, including Android, iOS, and web applications.

Data Flow Diagram



3. SYSTEM DESIGN – USER

CONTENT: **Welcome & Login page**
 User Registration Form

 Homepage
 History Transaction
 Profile

INTERFACE

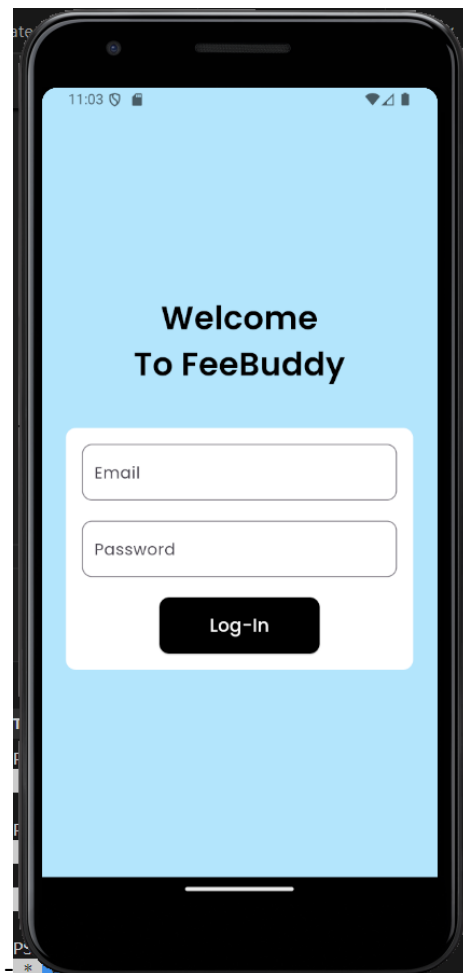
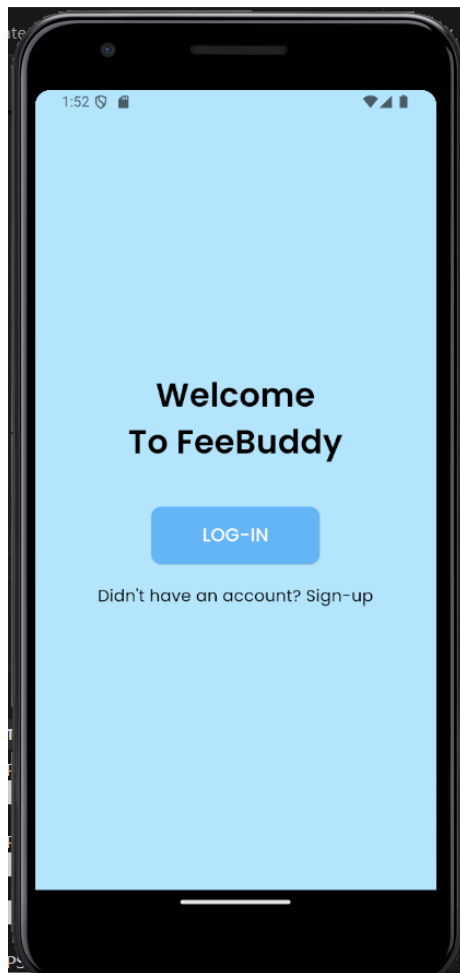
Welcome & Login Page:

This is the landing page of our application, where users will first arrive. The welcome page features a login panel that offers two options:

🌈 **Sign up** – for new users to create an account.

🌈 **Login** – for existing users to access their account and functionality.

This login panel serves as a gateway to our app's features and functionality.



User Registration Form:

After signing up, users will be directed to a registration form where they will provide their personal details, including:

- ✚ Full Name
- ✚ Email Address
- ✚ Password
- ✚ Semester
- ✚ Department

Upon submitting this form, the user's information will be automatically saved to our database, creating a unique profile for future access and payment processing.



Homepage:

After logging in, users arrive at the Home Page, featuring:

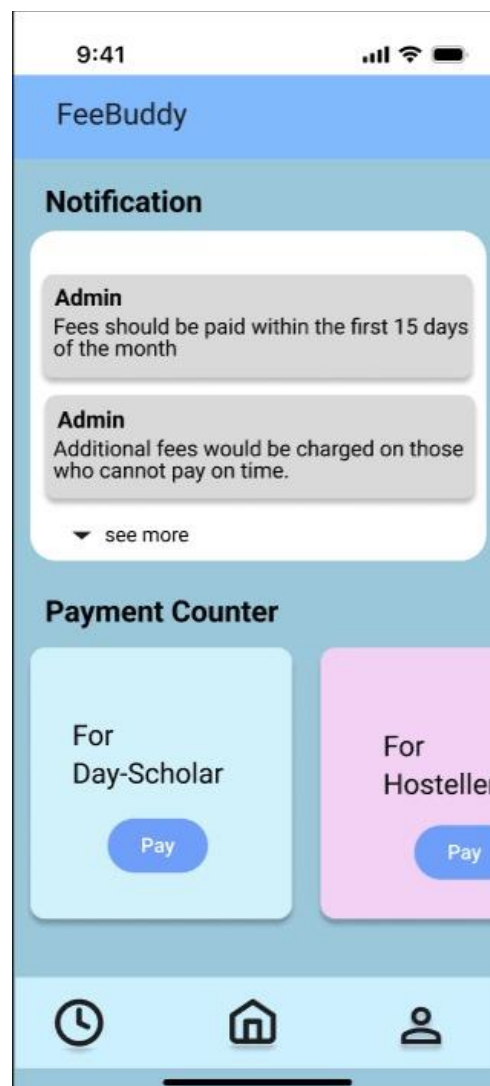
1. Fee Payment Calendar:

- ✚ Displays months with pending fee payments
- ✚ Allows users to select a month to pay fees

2. Payment Counter:

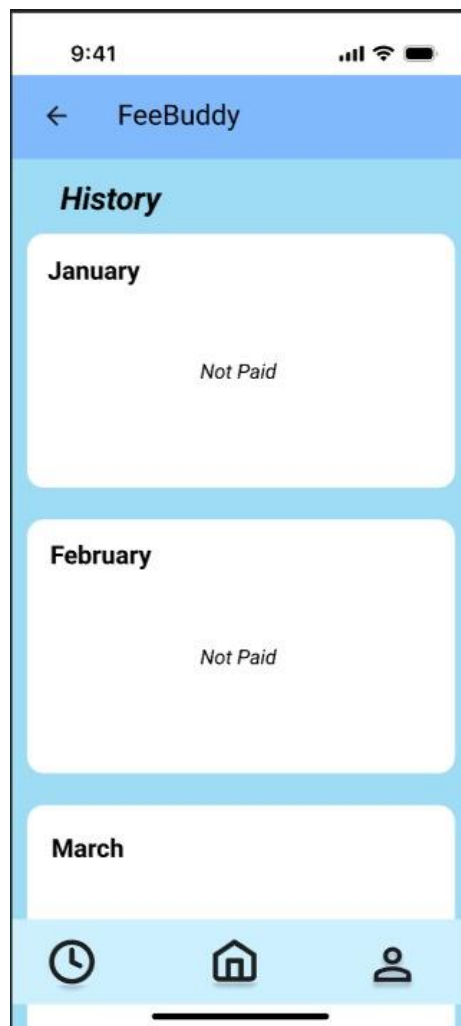
- ✚ Separate counters for Hostellers and Day Scholars
- ✚ Displays payment status and amount due

This page provides a clear overview of fee payment schedules and payment status, making it easy for users to manage their payments.



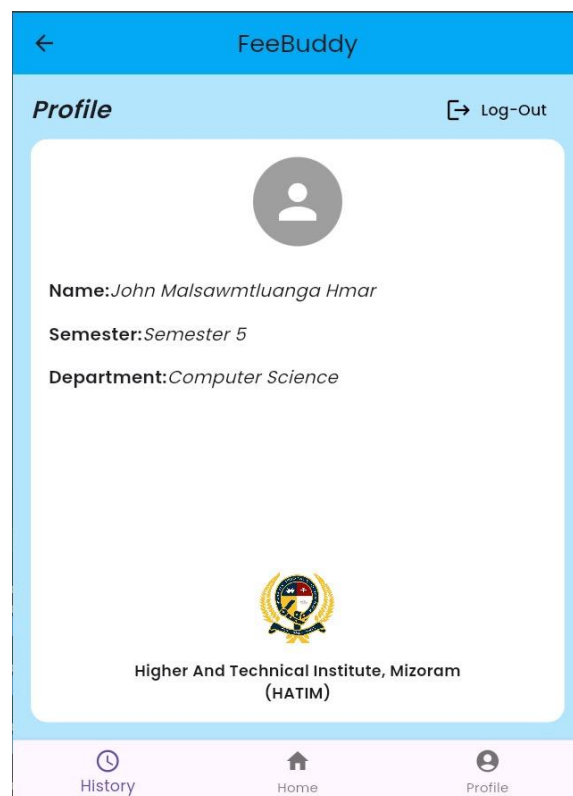
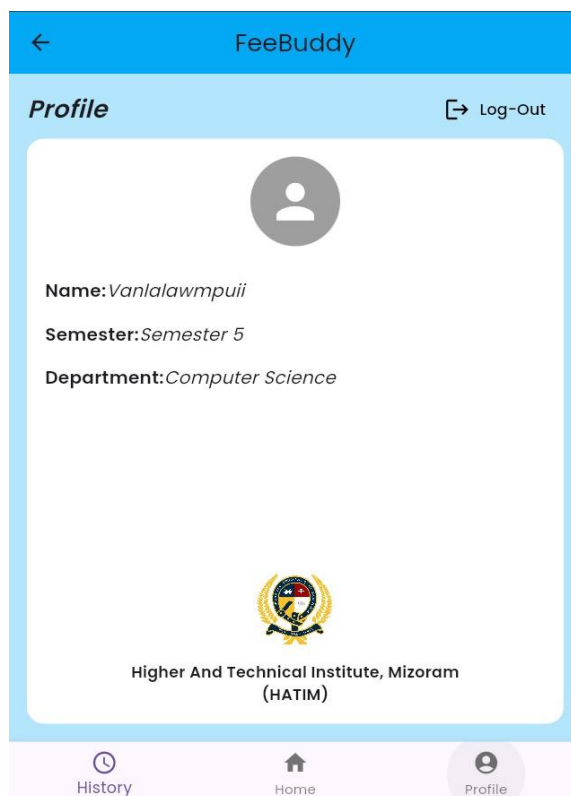
History Transaction:

The **Transaction History** page allows users or students to track their past payments and view the next scheduled payment month. It displays a detailed list of previous transactions, including payment dates, amounts, and statuses (e.g., "Paid" or "Pending"). With search and filter options for easy navigation, and secure access for data protection, this feature ensures efficient fee management and timely payment tracking.



Profile:

The **Profile Page** provides a comprehensive view of the student's details, displaying all the information submitted during the sign-up process. This includes the student's name, department, semester, and any other relevant details. Once the user logs in, the profile page automatically retrieves and displays these details from the database, ensuring that the student's information is readily accessible. This feature allows students to easily review and confirm their account information, ensuring accuracy and providing a personalized user experience.



4. SYSTEM DESIGN – ADMIN PANEL

CONTENT: Admin Home page

Students Table

Notification Page

Admin Home Page:

The admin homepage of the FeeBuddy app is designed to facilitate communication between the administrators and students. This interface includes the following features:

1. **Student Table Dropdown:**
 - ✚ A dropdown menu labeled "Student Table" allows the administrator to select a specific department. This selection ensures targeted notifications can be sent to students from a particular department.
2. **Notification Textbox:**
 - ✚ A text input field labeled "Notification" enables administrators to type custom messages. These messages can include reminders for fee payments, announcements, or other important updates relevant to the students.
3. **Upload Button:**
 - ✚ A blue "Upload" button is provided for submitting the composed notification. Once clicked, the notification is broadcasted to the selected group of students.
4. **Institute Branding:**
 - ✚ The app displays the logo and name of the institution, "Higher And Technical Institute, Mizoram (HATIM)," at the bottom of the screen to reinforce institutional identity.

The admin homepage is streamlined for efficient notification management, making it easier to handle student communication regarding fees and other updates.

The screenshot displays the 'FeeBuddy (Admin)' interface. At the top, there is a purple header bar with a back arrow and the text 'FeeBuddy (Admin)'. Below this, the 'Student Table:' section features a dropdown menu with the placeholder text 'Select the department' and a blue 'Select' button. The 'Notification:' section contains a large text input field with the placeholder 'Write here...' and a blue '+ Upload' button. At the bottom, the 'Uploaded Notifications:' section lists two notifications: 'Fees should be paid within the first 15 days of the month.' and 'Additional fees would be charged on those who cannot pay on time.', each with a red trash icon for deletion.

Students Table Page:

The "Student Table" screen in the FeeBuddy app provides administrators with an organized interface to manage student records. The features and layout are as follows:

1. **Header:**

- ✚ The page title "FeeBuddy (Admin)" is displayed at the top, providing clear context for the admin user.

2. **Student Table Section:**

- ✚ A table labeled "Department" displays student information in a structured format with the following columns:
 - **Name:** Shows the names of the students.
 - **Department:** Indicates the department to which each student belongs.
 - **Semester:** Displays the current semester of each student.

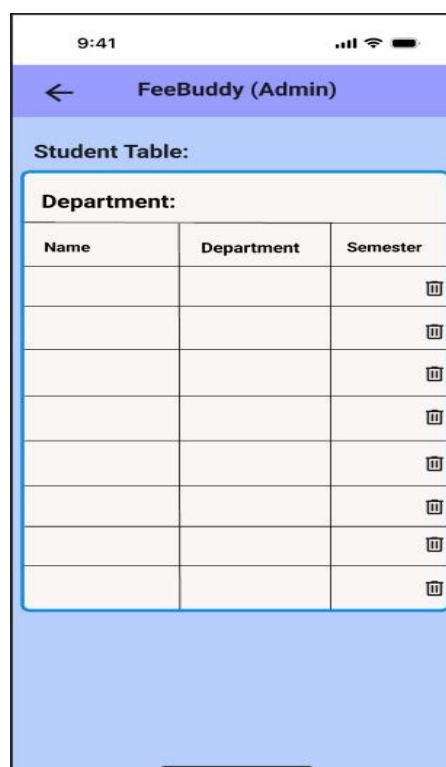
3. **Delete Option:**

- ✚ Each row includes a trash icon at the end, allowing administrators to delete individual student records as needed. This feature is useful for maintaining updated and accurate records.

4. **Scrollable List:**

- ✚ The table is designed to handle multiple rows, ensuring scalability for larger datasets.

This interface provides administrators with a streamlined method to view, update, and manage student data within the FeeBuddy app.





The screenshot displays the "Student Table" interface within the "FeeBuddy (Admin)" app. The table is titled "Department:" and has three columns: "Name", "Department", and "Semester". There are eight rows of data, each with a trash icon in the "Semester" column for deletion. The interface is shown on a mobile device screen with a status bar at the top showing the time as 9:41.

Name	Department	Semester
		🗑️
		🗑️
		🗑️
		🗑️
		🗑️
		🗑️
		🗑️
		🗑️


Notification Page:

The "**Notification Page**" screen in the FeeBuddy app allows administrators to create and review notifications sent to students. The screen is divided into two main sections:


1. Notification Input Section:

-  **Textbox:** A field labeled "Notification" where administrators can compose messages. This section is used for creating new announcements, reminders, or updates.
-  **Upload Button:** A blue button labeled "+ Upload" enables the administrator to submit the written notification.

2. Uploaded Notifications Section:

-  Displays a list of previously uploaded notifications for reference. Each notification is shown in a card-like format, with the following details:
 - **Sender Name:** The sender is labeled as "Admin," indicating the source of the message.
 - **Message Content:** The content of the notification is displayed below the sender's label. For instance, a sample notification states:
"Additional fees would be charged on those who cannot pay on time."

3. Back Navigation:

-  A back arrow is provided at the top-left corner to allow the admin to navigate to the previous screen.

This screen ensures efficient management of notifications, enabling the administrator to send timely updates and maintain a record of past communications.

4. CODING – USER

CONTENT:	Welcome & Login page
	User Registration Form
	Homepage
	History Transaction
	Profile

Welcome & Login page

Main

```
import 'package:feebuddy/pages/history.dart';
import 'package:feebuddy/pages/home.dart';
import 'package:feebuddy/pages/login.dart';
import 'package:feebuddy/pages/profile.dart';
import 'package:feebuddy/pages/signup.dart';
import 'package:feebuddy/pages/welcome.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:firebase_core/firebase_core.dart';
import 'package:flutter/foundation.dart';
import 'package:flutter/material.dart';

void main() async{
  WidgetsFlutterBinding.ensureInitialized();

  if(kIsWeb) {
    await Firebase.initializeApp(options: const FirebaseOptions(
      apiKey: "AlzaSyAr23UcOEXpNxulcyRYgqE1N2DEZSxOrkk",
      authDomain: "fee-buddy.firebaseio.com",
      projectId: "fee-buddy",
      storageBucket: "fee-buddy.appspot.com",
      messagingSenderId: "874905560826",
      appId: "1:874905560826:web:f5b8774644c4e9b3684b43",
      measurementId: "G-BRKVJEYZCL"));
  }else {
    await Firebase.initializeApp();
  }
}
```

```

}

runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  // This widget is the root of your application.

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: ThemeData(fontFamily: 'Poppins'),
      home: const AuthCheck(),
      routes: {
        '/signup': (context) => const SignUpScreen(),
        '/welcome': (context) => const WelScreen(),
        '/login': (context) => const LoginScreen(),
        '/home': (context) => const HomePage(),
        '/history': (context) => const HistoryScreen(),
        '/profile': (context) => const ProfileScreen(),
        // ignore: equal_keys_in_map
        '/welcome': (context) => const WelScreen()

        // Add other routes here if needed
      },
    );
  }
}

```

```

}

class AuthCheck extends StatelessWidget {

  const AuthCheck({super.key});

  @override

  Widget build(BuildContext context) {

    // Check if a user is currently logged in

    return StreamBuilder<User?>(

      stream: FirebaseAuth.instance.authStateChanges(), // Listen for auth state changes

      builder: (context, snapshot) {

        if (snapshot.connectionState == ConnectionState.waiting) {

          // While checking auth state, show a loading screen

          return const CircularProgressIndicator();

        } else if (snapshot.hasData) {

          // If a user is logged in, navigate to the home screen

          return const HomePage();

        } else {

          // If no user is logged in, show the welcome screen

          return const WelScreen();

        }

      },

    );

  }

}

```


Welcome page

```
import 'package:flutter/material.dart';

class WelScreen extends StatelessWidget {

  const WelScreen({super.key});

  @override
  Widget build(BuildContext context) {

    return Scaffold(

      backgroundColor: const Color(0xFFB3E5FC), // Light blue background color

      body: Center(

        child: Padding(

          padding: const EdgeInsets.symmetric(horizontal: 24.0),

          child: Column(

            mainAxisAlignment: MainAxisAlignment.center,

            children: [

              const Text(

                'Welcome\nTo FeeBuddy',

                textAlign: TextAlign.center,

                style: TextStyle(

                  fontSize: 32,

                  fontWeight: FontWeight.bold,

                  color: Colors.black,

                ),

              ),

              const SizedBox(height: 40),

              ElevatedButton(
```

```

onPressed: () {
  // Navigate to the login page
  Navigator.pushNamed(context, '/login');
},
style: ElevatedButton.styleFrom(
  backgroundColor: const Color(0xFF64B5F6), // Button background color
  shape: RoundedRectangleBorder(
    borderRadius: BorderRadius.circular(10),
  ),
  padding: const EdgeInsets.symmetric(
    horizontal: 50,
    vertical: 15,
  ),
),
child: const Text(
  'LOG-IN',
  style: TextStyle(
    fontSize: 18,
    color: Colors.white,
  ),
),
),
const SizedBox(height: 20),
GestureDetector(
  onTap: () {
    Navigator.pushNamed(context, '/signup');
  },
),

```

```
        // Navigate to the sign-up page
      },
      child: const Text(
        "Didn't have an account? Sign-up",
        style: TextStyle(
          fontSize: 16,
          color: Colors.black,
        ),
      ),
    ),
  ],
),
),
);
}
```

Login

```
import 'package:feebuddy/auth/auth_service.dart';

import 'package:flutter/material.dart';

import 'dart:developer';

class LoginScreen extends StatefulWidget {

  const LoginScreen({super.key});

  @override

  State<LoginScreen> createState() => _LoginScreenState();

}

class _LoginScreenState extends State<LoginScreen> {

  final _auth = AuthService();

  final email = TextEditingController();

  final password = TextEditingController();

  @override

  void dispose() {

    email.dispose();

    password.dispose();

    super.dispose();

  }

  Future<void> _login() async{
```

```

try {
    final user = await _auth.loginUserWithEmailAndPassword(email.text, password.text);
    if(user!=null) {
        log("User Logged In Successfully!");
        // ignore: use_build_context_synchronously
        Navigator.pushNamed(context, '/home');
    }
} catch(e) {
    log("Invalid Email or Password!");
}
}

```

@override

```

Widget build(BuildContext context) {
    return Scaffold(
        backgroundColor: const Color(0xFFB3E5FC), // Light blue background color
        body: Center(
            child: SingleChildScrollView(
                padding: const EdgeInsets.symmetric(horizontal: 24.0),
                child: Column(
                    mainAxisAlignment: MainAxisAlignment.center,
                    children: [
                        const Text(
                            'Welcome\nTo FeeBuddy',
                            textAlign: TextAlign.center,
                            style: TextStyle(

```

```

    fontSize: 32,
    fontWeight: FontWeight.bold,
    color: Colors.black,
  ),
),
const SizedBox(height: 40),
Container(
  padding: const EdgeInsets.all(16.0),
  decoration: BoxDecoration(
    color: Colors.white,
    borderRadius: BorderRadius.circular(10),
  ),
  child: Column(
    children: [
      TextField(
        controller: email,
        decoration: InputDecoration(
          labelText: 'Email',
          hintText: 'email',
          border: OutlineInputBorder(
            borderRadius: BorderRadius.circular(10),
          ),
        ),
      ),
    ],
  ),
  const SizedBox(height: 20),
  TextField(

```

```

controller: password,
decoration: InputDecoration(
  labelText: 'Password',
  hintText: 'password',
  border: OutlineInputBorder(
    borderRadius: BorderRadius.circular(10),
  ),
),
obscureText: true,
),
const SizedBox(height: 20),
ElevatedButton(
  onPressed: _login,
  style: ElevatedButton.styleFrom(
    backgroundColor: Colors.black,
    shape: RoundedRectangleBorder(
      borderRadius: BorderRadius.circular(10),
    ),
    padding: const EdgeInsets.symmetric(
      horizontal: 50,
      vertical: 15,
    ),
  ),
  child: const Text(
    'Log-In',
    style: TextStyle(

```

```
        fontSize: 18,  
        color: Colors.white,  
      ),  
    ),  
  ),  
  ],  
),  
),  
  ],  
  ),  
  ),  
  ),  
);  
}  
}
```


User Registration Form

```
import 'package:feebuddy/auth/auth_service.dart';

import 'package:flutter/material.dart';

import 'dart:developer';

class LoginScreen extends StatefulWidget {

  const LoginScreen({super.key});

  @override

  State<LoginScreen> createState() => _LoginScreenState();
}

class _LoginScreenState extends State<LoginScreen> {

  final _auth = AuthService();

  final email = TextEditingController();

  final password = TextEditingController();

  @override

  void dispose() {

    email.dispose();

    password.dispose();

    super.dispose();

  }

  Future<void> _login() async{
```

```

try {
    final user = await _auth.loginUserWithEmailAndPassword(email.text, password.text);
    if(user!=null) {
        log("User Logged In Successfully!");
        // ignore: use_build_context_synchronously
        Navigator.pushNamed(context, '/home');
    }
} catch(e) {
    log("Invalid Email or Password!");
}
}

```

@override

```

Widget build(BuildContext context) {
    return Scaffold(
        backgroundColor: const Color(0xFFB3E5FC), // Light blue background color
        body: Center(
            child: SingleChildScrollView(
                padding: const EdgeInsets.symmetric(horizontal: 24.0),
                child: Column(
                    mainAxisAlignment: MainAxisAlignment.center,
                    children: [
                        const Text(
                            'Welcome\nTo FeeBuddy',
                            textAlign: TextAlign.center,
                            style: TextStyle(

```

```

    fontSize: 32,

    fontWeight: FontWeight.bold,

    color: Colors.black,

  ),

),

const SizedBox(height: 40),

Container(

  padding: const EdgeInsets.all(16.0),

  decoration: BoxDecoration(

    color: Colors.white,

    borderRadius: BorderRadius.circular(10),

  ),

  child: Column(

    children: [

      TextField(

        controller: email,

        decoration: InputDecoration(

          labelText: 'Email',

          hintText: 'email',

          border: OutlineInputBorder(

            borderRadius: BorderRadius.circular(10),

          ),

        ),

      ),

    ],

  ),

  const SizedBox(height: 20),

  TextField(

```

```

controller: password,
decoration: InputDecoration(
  labelText: 'Password',
  hintText: 'password',
  border: OutlineInputBorder(
    borderRadius: BorderRadius.circular(10),
  ),
),
obscureText: true,
),
const SizedBox(height: 20),
ElevatedButton(
  onPressed: _login,
  style: ElevatedButton.styleFrom(
    backgroundColor: Colors.black,
    shape: RoundedRectangleBorder(
      borderRadius: BorderRadius.circular(10),
    ),
    padding: const EdgeInsets.symmetric(
      horizontal: 50,
      vertical: 15,
    ),
  ),
  child: const Text(
    'Log-In',
    style: TextStyle(

```

```
        fontSize: 18,  
        color: Colors.white,  
      ),  
    ),  
  ),  
  ],  
),  
),  
  ],  
  ),  
  ),  
  ),  
);  
}  
}
```

Homepage

```
import 'package:flutter/material.dart';

class HomePage extends StatefulWidget {
  const HomePage({super.key});

  @override
  // ignore: library_private_types_in_public_api
  _HomeScreenState createState() => _HomeScreenState();
}

class _HomeScreenState extends State<HomePage> {
  int _selectedIndex = 0;

  void _onItemTapped(int index) {
    setState(() {
      _selectedIndex = index;
    });

    switch (index) {
      case 0:
        Navigator.pushNamed(context, '/history');
        break;
      case 1:
        Navigator.pushNamed(context, '/home');
        break;
      case 2:
        Navigator.pushNamed(context, '/profile');
        break;
    }
  }
}
```

```

}

@override
Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: const Color(0xFFB3E5FC),
    appBar: AppBar(
      backgroundColor: Colors.lightBlue,
      title: const Text('FeeBuddy'),
      centerTitle: true,
      elevation: 0,
    ),
    body: Padding(
      padding: const EdgeInsets.all(16.0),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          // Notifications Section
          const Text(
            'Notification',
            style: TextStyle(fontSize: 18, fontWeight: FontWeight.bold),
          ),
          const SizedBox(height: 10),
          Container(
            padding: const EdgeInsets.all(10.0),
            decoration: BoxDecoration(
              color: Colors.lightBlue[100],

```

```
borderRadius: BorderRadius.circular(15),
),
child: Column(
  children: [
    const NotificationCard(
      title: 'Admin',
      message: 'Fees should be paid within the first 15 days of the month.',
    ),
    const SizedBox(height: 10),
    const NotificationCard(
      title: 'Admin',
      message: 'Additional fees would be charged on those who cannot pay on time.',
    ),
  ],
),
Row(
  mainAxisAlignment: MainAxisAlignment.end,
  children: [
    TextButton.icon(
      onPressed: () {},
      icon: const Icon(Icons.arrow_drop_down),
      label: const Text("see more"),
    ),
  ],
),
),
),
```



```

const SizedBox(height: 20),

// Payment Counter Section
const Text(
  'Payment Counter',
  style: TextStyle(fontSize: 18, fontWeight: FontWeight.bold),
),
const SizedBox(height: 10),
Row(
  mainAxisAlignment: MainAxisAlignment.spaceAround,
  children: [
    PaymentCard(
      label: "For Day-Scholar",
      color: Colors.lightBlue[100]!,
    ),
    PaymentCard(
      label: "For Hosteller",
      color: Colors.pink[100]!,
    ),
  ],
),
],
),
),
bottomNavigationBar: BottomNavigationBar(

```

```

        currentIndex: _selectedIndex,
        onTap: _onItemTapped,
        items: const [
            BottomNavigationBarItem(
                icon: Icon(Icons.access_time),
                label: 'History',
            ),
            BottomNavigationBarItem(
                icon: Icon(Icons.home),
                label: 'Home',
            ),
            BottomNavigationBarItem(
                icon: Icon(Icons.account_circle),
                label: 'Profile',
            ),
        ],
    ),
);
}
}

// Notification Card Widget
class NotificationCard extends StatelessWidget {
    final String title;
    final String message;

```

```

const NotificationCard({
  super.key,
  required this.title,
  required this.message,
});

@override
Widget build(BuildContext context) {
  return Card(
    elevation: 3,
    child: ListTile(
      leading: const CircleAvatar(
        backgroundColor: Colors.black,
        child: Icon(Icons.admin_panel_settings, color: Colors.white),
      ),
      title: Text(
        title,
        style: const TextStyle(fontWeight: FontWeight.bold),
      ),
      subtitle: Text(message),
    ),
  );
}

// Payment Card Widget

```

```

class PaymentCard extends StatelessWidget {

  final String label;

  final Color color;

  const PaymentCard({super.key, required this.label, required this.color});

  @override
  Widget build(BuildContext context) {
    return Container(
      padding: const EdgeInsets.all(16.0),
      width: MediaQuery.of(context).size.width * 0.4,
      decoration: BoxDecoration(
        color: color,
        borderRadius: BorderRadius.circular(15),
        boxShadow: const [
          BoxShadow(
            color: Colors.black26,
            blurRadius: 5,
            offset: Offset(2, 2),
          ),
        ],
      ),
      child: Column(
        children: [
          Text(
            label,

```

```

    textAlign: TextAlign.center,

    style: const TextStyle(

      fontSize: 16,

      fontWeight: FontWeight.bold,

      color: Colors.black87,

    ),

  ),

  const SizedBox(height: 10),

  ElevatedButton(

    onPressed: () {

      // Add payment functionality

    },

    style: ElevatedButton.styleFrom(

      backgroundColor: Colors.blueAccent,

      shape: RoundedRectangleBorder(

        borderRadius: BorderRadius.circular(20),

      ),

    ),

    child: const Text("Pay"),

  ),

],

),

);

}

}

```

History Transaction

```
import "package:flutter/material.dart";

class HistoryScreen extends StatefulWidget {
  const HistoryScreen({super.key});

  @override
  // ignore: library_private_types_in_public_api
  _HistoryScreenState createState() => _HistoryScreenState();
}

class _HistoryScreenState extends State<HistoryScreen> {
  int _selectedIndex = 0;

  void _onItemTapped(int index) {
    setState(() {
      _selectedIndex = index;
    });

    switch (index) {
      case 0:
        Navigator.pushNamed(context, '/history');
        break;
      case 1:
        Navigator.pushNamed(context, '/home');
        break;
    }
  }
}
```

case 2:

```
    Navigator.pushNamed(context, '/profile');

    break;
}

}

// Sample data for history
final List<Map<String, String>> _paymentHistory = [

    {"month": "January", "status": "Not Paid"},
    {"month": "February", "status": "Not Paid"},
    {"month": "March", "status": "Not Paid"},
    {"month": "April", "status": "Not Paid"},
    {"month": "May", "status": "Not Paid"},
    {"month": "June", "status": "Not Paid"},
    {"month": "July", "status": "Not Paid"},
    {"month": "August", "status": "Not Paid"},
    {"month": "September", "status": "Not Paid"},
    {"month": "October", "status": "Not Paid"},
    {"month": "November", "status": "Not Paid"},
    {"month": "December", "status": "Not Paid"},

];
```

@override

```
Widget build(BuildContext context) {

    return Scaffold(

        backgroundColor: const Color(0xFFB3E5FC),
```

```

appBar: AppBar(
  backgroundColor: Colors.lightBlue,
  title: const Text('FeeBuddy'),
  leading: IconButton(
    icon: const Icon(Icons.arrow_back),
    onPressed: () {
      // Add logic to navigate back
      Navigator.pop(context);
    },
  ),
  centerTitle: true,
  elevation: 0,
),
body: Padding(
  padding: const EdgeInsets.all(16.0),
  child: Column(
    crossAxisAlignment: CrossAxisAlignment.start,
    children: [
      const Text(
        'History',
        style: TextStyle(
          fontSize: 22,
          fontWeight: FontWeight.bold,
          fontStyle: FontStyle.italic,
        ),
      ),
    ],
  ),
),

```



```

const SizedBox(height: 10),
Expanded(
  child: ListView.builder(
    itemCount: _paymentHistory.length,
    itemBuilder: (context, index) {
      return Container(
        margin: const EdgeInsets.only(bottom: 10),
        padding: const EdgeInsets.all(16.0),
        decoration: BoxDecoration(
          color: Colors.lightBlue[100],
          borderRadius: BorderRadius.circular(15),
        ),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            Text(
              _paymentHistory[index]["month"]!,
              style: const TextStyle(
                fontSize: 18,
                fontWeight: FontWeight.bold,
              ),
            ),
            const SizedBox(height: 5),
            Text(
              _paymentHistory[index]["status"]!,
              style: const TextStyle(

```

```

        fontSize: 16,
        fontStyle: FontStyle.italic,
      ),
    ),
  ],
),
);
},
),
),
],
),
),
bottomNavigationBar: BottomNavigationBar(
  currentIndex: _selectedIndex,
  onTap: _onItemTapped,
  items: const [
    BottomNavigationBarItem(
      icon: Icon(Icons.access_time),
      label: 'History',
    ),
    BottomNavigationBarItem(
      icon: Icon(Icons.home),
      label: 'Home',
    ),
    BottomNavigationBarItem(

```

```
        icon: Icon(Icons.account_circle),  
        label: 'Profile',  
      ),  
    ],  
  ),  
);  
}  
}
```

Profile

```
import 'dart:math';

import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';

class ProfileScreen extends StatefulWidget {
  const ProfileScreen({super.key});

  @override
  // ignore: library_private_types_in_public_api
  _ProfileScreenState createState() => _ProfileScreenState();
}

class _ProfileScreenState extends State<ProfileScreen> {
  int _selectedIndex = 0;

  String name = '\t';
  String semester = '\t';
  String department = '\t';

  @override
  void initState() {
    super.initState();

    _fetchUserInfo(); // Fetch user info when screen loads
  }
}
```

```

Future<void> _fetchUserInfo() async {
  try {
    // Get current user from FirebaseAuth
    User? currentUser = FirebaseAuth.instance.currentUser;

    if (currentUser != null) {
      // Fetch user document from Firestore
      DocumentSnapshot userDoc = await FirebaseFirestore.instance
        .collection('users')
        .doc(currentUser.uid)
        .get();

      if (userDoc.exists) {
        // Extract user data
        setState(() {
          name = userDoc['name'];
          semester = userDoc['semester'];
          department = userDoc['department'];
        });
      }
    }
  } catch (e) {
    log("Error fetching user data: $e" as num);
  }
}

```

```

void _onItemTapped(int index) {
  setState(() {
    _selectedIndex = index;
  });

  switch (index) {
    case 0:
      Navigator.pushNamed(context, '/history');
      break;
    case 1:
      Navigator.pushNamed(context, '/home');
      break;
    case 2:
      Navigator.pushNamed(context, '/profile');
      break;
  }
}

```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: const Color(0xFFB3E5FC),
    appBar: AppBar(
      backgroundColor: Colors.lightBlue,
      title: const Text('FeeBuddy'),
      leading: IconButton(

```

```

        icon: const Icon(Icons.arrow_back),
        onPressed: () {
          // Navigate back
          Navigator.pop(context);
        },
      ),
      centerTitle: true,
      elevation: 0,
    ),
    body: Padding(
      padding: const EdgeInsets.all(16.0),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          Row(
            mainAxisAlignment: MainAxisAlignment.spaceBetween,
            children: [
              const Text(
                'Profile',
                style: TextStyle(
                  fontSize: 22,
                  fontWeight: FontWeight.bold,
                  fontStyle: FontStyle.italic,
                ),
              ),
              TextButton.icon(

```

```

    onPressed: () async {
      await FirebaseAuth.instance.signOut(); // Sign out the user
      // ignore: use_build_context_synchronously
      Navigator.pushReplacementNamed(context, '/welcome'); // Redirect to welcome
screen after logout
    },

    icon: const Icon(Icons.logout, color: Colors.black),
    label: const Text(
      'Log-Out',
      style: TextStyle(color: Colors.black),
    ),
  ),
],
),
const SizedBox(height: 10),
Expanded(
  child: Container(
    padding: const EdgeInsets.all(16.0),
    decoration: BoxDecoration(
      color: Colors.white,
      borderRadius: BorderRadius.circular(15),
    ),
    child: Column(
      children: [
        const CircleAvatar(

```



```

radius: 40,

backgroundColor: Colors.grey,

child: Icon(

  Icons.person,

  size: 50,

  color: Colors.white,

),

),

const SizedBox(height: 20),

// Display fetched profile information

buildProfileField('Name:', name),

buildProfileField('Semester:', semester),

buildProfileField('Department:', department),

const SizedBox(height: 20),

Expanded(

  child: Column(

    mainAxisAlignment: MainAxisAlignment.end,

    children: [

      Image.asset(

        'assets/icons/hatim_logo.png', // Replace this with your logo path

        height: 70,

      ),

      const SizedBox(height: 10),

      const Text(

        'Higher And Technical Institute, Mizoram\n(HATIM)',

        textAlign: TextAlign.center,

```



```

    ),
    BottomNavigationBarItem(
      icon: Icon(Icons.account_circle),
      label: 'Profile',
    ),
  ],
),
);
}

```

// Method to build each profile field with the fetched data

```

Widget buildProfileField(String label, String value) {
  return Padding(
    padding: const EdgeInsets.symmetric(vertical: 8.0),
    child: Row(
      mainAxisAlignment: MainAxisAlignment.start,
      children: [
        Text(
          label,
          style: const TextStyle(
            fontSize: 16,
            fontWeight: FontWeight.bold,
          ),
        ),
        Text(
          value.isEmpty ? '_' : value,

```

```
style: const TextStyle(  
  fontSize: 16,  
  fontStyle: FontStyle.italic,  
),  
),  
],  
),  
);  
}  
}
```

6. CODING – ADMIN PANEL

**CONTENT: Admin Home page &
Notification Page
Students Table**

Admin Home page & Notification Page

```
// ignore_for_file: library_private_types_in_public_api, use_build_context_synchronously
```

```
import 'package:flutter/material.dart';
```

```
import 'package:cloud_firestore/cloud_firestore.dart'; // Firebase Firestore
```

```
import 'adminstudenttable.dart'; // Ensure you have this file for the next page
```

```
class AdminPanel extends StatefulWidget {
```

```
  const AdminPanel({super.key});
```

```
  @override
```

```
  _AdminPanelState createState() => _AdminPanelState();
```

```
}
```

```
class _AdminPanelState extends State<AdminPanel> {
```

```
  String? selectedDepartment; // To store the selected department
```

```
  final TextEditingController notificationController = TextEditingController();
```

```
  void uploadNotification() async {
```

```
    debugPrint('Uploading notification...');
```

```
    if (notificationController.text.trim().isEmpty) {
```

```
      debugPrint('Notification text is empty.');
```

```
      ScaffoldMessenger.of(context).showSnackBar(
```

```
        const SnackBar(content: Text('Notification text cannot be empty')),
```

```
      );
```

```
      return;
```

```

}

try {
  // Save notification to Firestore
  await FirebaseFirestore.instance.collection('notifications').add({
    'text': notificationController.text.trim(),
    'timestamp': FieldValue.serverTimestamp(),
  }).then((value) {
    debugPrint('Notification uploaded: ${value.id}');
  }).catchError((error) {
    debugPrint('Failed to upload notification: $error');
  });

  // Clear the input field
  notificationController.clear();

  ScaffoldMessenger.of(context).showSnackBar(
    const SnackBar(content: Text('Notification uploaded successfully')),
  );
} catch (e) {
  ScaffoldMessenger.of(context).showSnackBar(
    SnackBar(content: Text('Failed to upload notification: $e')),
  );
}
}

```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: const Text('FeeBuddy (Admin)'),
      backgroundColor: const Color(0xFF9575CD),
    ),
    body: Padding(
      padding: const EdgeInsets.all(16.0),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          const Text(
            'Student Table:',
            style: TextStyle(fontSize: 18, fontWeight: FontWeight.bold),
          ),
          const SizedBox(height: 10),
          DropdownButtonFormField<String>(
            decoration: InputDecoration(
              filled: true,
              fillColor: Colors.white,
              border: OutlineInputBorder(
                borderRadius: BorderRadius.circular(10),
              ),
            ),
            hint: const Text('Select the department'),

```



```

items: const [
  DropdownMenuItem(value: 'Commerce', child: Text('Commerce')),
  DropdownMenuItem(value: 'Computer Science', child: Text('Computer Science')),
  DropdownMenuItem(value: 'Education', child: Text('Education')),
  DropdownMenuItem(value: 'English', child: Text('English')),
  DropdownMenuItem(value: 'History', child: Text('History')),
  DropdownMenuItem(value: 'Philosophy', child: Text('Philosophy')),
  DropdownMenuItem(value: 'Psychology', child: Text('Psychology')),
  DropdownMenuItem(value: 'Social Work', child: Text('Social Work')),
],
onChanged: (value) {
  setState(() {
    selectedDepartment = value;
  });
},
),
const SizedBox(height: 10),
Center(
  child: ElevatedButton(
    style: ElevatedButton.styleFrom(
      backgroundColor: const Color(0xFF64B5F6),
      padding: const EdgeInsets.symmetric(vertical: 10, horizontal: 50),
    ),
    onPressed: () {
      if (selectedDepartment != null) {
        Navigator.push(

```

```

        context,

        MaterialPageRoute(

            builder: (context) => AdminStudentTable(department: selectedDepartment!),

        ),

    );

    } else {

        ScaffoldMessenger.of(context).showSnackBar(

            const SnackBar(content: Text('Please select a department')),

        );

    }

    },

    child: const Text('Select'),

    ),

    ),

    const SizedBox(height: 20),

    const Text(

        'Notification:',

        style: TextStyle(fontSize: 18, fontWeight: FontWeight.bold),

    ),

    const SizedBox(height: 10),

    TextField(

        controller: notificationController,

        maxLines: 4,

        decoration: InputDecoration(

            filled: true,

            fillColor: Colors.white,

```

```

    hintText: 'Write here...',

    border: OutlineInputBorder(
      borderRadius: BorderRadius.circular(10),
    ),
  ),
),
const SizedBox(height: 10),
Center(
  child: ElevatedButton(
    style: ElevatedButton.styleFrom(
      backgroundColor: const Color(0xFF64B5F6),
      padding: const EdgeInsets.symmetric(vertical: 10, horizontal: 50),
    ),
    onPressed: uploadNotification,
    child: const Text('+ Upload'),
  ),
),
const SizedBox(height: 20),
const Text(
  'Uploaded Notifications:',
  style: TextStyle(fontSize: 18, fontWeight: FontWeight.bold),
),
const SizedBox(height: 10),
Expanded(
  child: StreamBuilder<QuerySnapshot>(
    stream: FirebaseFirestore.instance

```

```

.collection('notifications')

.orderBy('timestamp', descending: true)

.snapshots(),
builder: (context, snapshot) {

  if (snapshot.connectionState == ConnectionState.waiting) {

    return const Center(child: CircularProgressIndicator());

  }

  if (!snapshot.hasData || snapshot.data!.docs.isEmpty) {

    return const Center(child: Text('No notifications uploaded yet.'));

  }

  return ListView(

    children: snapshot.data!.docs.map((doc) {

      final notification = doc['text'];

      return Card(

        margin: const EdgeInsets.symmetric(vertical: 5),

        child: ListTile(

          title: Text(notification),

          trailing: IconButton(

            icon: const Icon(Icons.delete, color: Colors.red),

            onPressed: () async {

              await doc.reference.delete();

              ScaffoldMessenger.of(context).showSnackBar(

                const SnackBar(content: Text('Notification deleted')),

              );

            },

```

```
        ),
        ),
    );
    }).toList(),
    );
    },
    ),
    ),
    ],
    ),
    ),
    backgroundColor: const Color(0xFFE3F2FD),
  );
}
}
```

Students Table

```
import 'package:flutter/material.dart';

import 'package:cloud_firestore/cloud_firestore.dart';

class AdminStudentTable extends StatelessWidget {

  final String department; // The selected department passed from the Admin Panel

  const AdminStudentTable({super.key, required this.department});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('FeeBuddy (Admin)'),
        backgroundColor: const Color(0xFF9575CD),
        leading: IconButton(
          icon: const Icon(Icons.arrow_back),
          onPressed: () {
            Navigator.pop(context); // Navigate back to the Admin Panel
          },
        ),
      ),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
```

```

children: [
  const Text(
    'Student Table:',
    style: TextStyle(fontSize: 18, fontWeight: FontWeight.bold),
  ),
  const SizedBox(height: 10),
  Text(
    'Department: $department',
    style: const TextStyle(fontSize: 16),
  ),
  const SizedBox(height: 20),
  Expanded(
    child: StreamBuilder<QuerySnapshot>(
      stream: FirebaseFirestore.instance
        .collection('users') // Ensure you have a 'users' collection
        .where('department', isEqualTo: department)
        .orderBy('semester', descending: true) // Sort by semester
        .snapshots(),
      builder: (context, snapshot) {
        if (snapshot.connectionState == ConnectionState.waiting) {
          return const Center(child: CircularProgressIndicator());
        }
        if (snapshot.hasError) {
          return Center(child: Text('Error: ${snapshot.error}'));
        }
        debugPrint('Snapshot data: ${snapshot.data}');
      },
    ),
  ),
],

```

```

if (!snapshot.hasData || snapshot.data!.docs.isEmpty) {
    return const Center(child: Text('No students found.'));
}

final students = snapshot.data!.docs;

return SingleChildScrollView(
    scrollDirection: Axis.horizontal, // Enable horizontal scrolling
    child: DataTable(
        border: TableBorder.all(color: Colors.grey),
        columns: const [
            DataColumn(label: Text('Name')),
            DataColumn(label: Text('Department')),
            DataColumn(label: Text('Semester')),
            DataColumn(label: Text('Actions')),
        ],
        rows: students.map((doc) {
            final name = doc['name'];
            final department = doc['department'];
            final semester = doc['semester'];

            return DataRow(cells: [
                DataCell(Text(name)),
                DataCell(Text(department)),
                DataCell(Text(semester.toString())),
                DataCell(

```



```

IconButton(
  icon: const Icon(Icons.delete, color: Colors.red),
  onPressed: () async {
    await doc.reference.delete();

    // ignore: use_build_context_synchronously
    ScaffoldMessenger.of(context).showSnackBar(
      const SnackBar(content: Text('Student deleted')),
    );
  },
),
),
]);
}).toList(),
),
);
},
),
),
],
),
),
backgroundColor: const Color(0xFFE3F2FD),
);
}
}

```